

Progressive Coding for Hyperspectral Signature Characterization

Chein-I Chang¹ Jing Wang¹ Chein-Chi Chang² Chinsu Lin³

¹Remote Sensing Signal and Image Processing Laboratory

Department of Computer Science and Electrical Engineering

University of Maryland Baltimore County, Baltimore, MD 21250

²Department of Civil and Environmental Engineering

University of Maryland Baltimore County, Baltimore, MD 21250

³Environment Monitoring and Planning Laboratory

Department of Forestry, National Chiayi University (NCYU)

300 University Road, Chiayi (600) Taiwan

Report Documentation Page

*Form Approved
OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE 2006	2. REPORT TYPE	3. DATES COVERED 00-00-2006 to 00-00-2006	
4. TITLE AND SUBTITLE Progressive Coding for Hyperspectral Signature Characterization		5a. CONTRACT NUMBER	
		5b. GRANT NUMBER	
		5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)		5d. PROJECT NUMBER	
		5e. TASK NUMBER	
		5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) University of Maryland, Baltimore County, Department of Computer Science and Electrical Engineering, Baltimore, MD, 21228		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)	
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited			
13. SUPPLEMENTARY NOTES The original document contains color images.			
14. ABSTRACT			
15. SUBJECT TERMS			
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified	
			18. NUMBER OF PAGES 43
			19a. NAME OF RESPONSIBLE PERSON

ABSTRACT

Spectral signature coding is an effective means of characterizing spectral features. This paper develops a rather different encoding concept, called progressive signature coding (PSC) which encodes a signature in a hierarchical manner. More specifically, it progressively encodes a spectral signature in multiple stages, each of these stages captures disjoint spectral information contained in the spectral signature. As a result of such a progressive coding, a spectral profile of progressive changes in a spectral signature can be generated for spectral characterization. The proposed idea is very simple and evolved from the Pulse Code Modulation (PCM) commonly used in communications and signal processing. It expands PCM to multi-stage PCM (MPCM) in the sense that a signature can be decomposed and quantized by PCM progressively in multiple stages for spectral characterization. In doing so, the MPCM generates a priority code for a spectral signature so that its spectral information captured in different stages can be prioritized in accordance with significance of changes in spectral variation. Such MPCM-based progressive spectral signature coding (MPCM-PSSC) can be useful in applications such as hyperspectral data exploitation, environmental monitoring, chemical/biological agent detection. Experiments are provided to demonstrate the utility of the MPCM-PSSC in signature discrimination and identification.

Subject Terms: Multistage Pulse Coding Modulation (MPCM). MPCM-based progressive spectral signature coding (MPCM-PSSC). Progressive spectral signature coding (PSSC). Spectral signature coding (SSC). Spectral discrimination. Spectral identification.

1. INTRODUCTION

Spectral signature coding (SSC) is a scheme or a rule or a mapping that transforms spectral values into a new set of symbols in a specific manner that a signature can be represented by the new symbols more effectively or efficiently. In hyperspectral imagery, each pixel is acquired by hundreds of contiguous spectral channels to form a column vector that can be used to diagnose subtle material substances based on their spectral characteristics. Therefore, taking advantage of such intra-pixel spectral information (e.g., spectral information provided by spectral channels within a hyperspectral image pixel vector) is one of great benefits resulting from hyperspectral data. However, this also is traded off for a price that many unknown spectral signatures may be also extracted to further complicate spectral analysis. So, one of major challenges in hyperspectral data exploitation is how to best utilize the spectral information provided by hyperspectral imagery to accomplish tasks such as detection, discrimination, classification, identification while discarding undesired information caused by unwanted interference such as noise.

This paper investigates a new approach to SSC, called progressive spectral signature coding (PSSC) where the SSC is carried out in a progressive fashion rather than sequential coding. It is a technique that can decompose a signature in multiple stages, each of these stages captures spectral changes in a progressive manner.¹⁻³ As a consequence, it provides a profile of progressive changes in spectral variation that describes the spectral behavior of a pixel vector in various stages. Accordingly, we can consider PSSC as “soft” coding in a progressive procedure as opposed to SSC that can be viewed as “hard” coding performed by classical coding techniques with binary decisions.

This paper develops a technique, called Multistage Pulse Coding Modulation (MPCM) for

PSSC, (MPCM-PSSC) which was previously developed for progressive image compression.⁴ The success of MPCM has been also demonstrated in image progressive reconstruction,⁵⁻⁶ progressive edge detection⁷⁻⁸ and progressive text detection.⁹ It is indeed a one-dimensional (1-D) transform coding technique, which encodes a 1-D signal progressively according to a priority assigned to each signal point. The signal priorities are determined by changes between two successive signal points. Interestingly, a hyperspectral image pixel can be also considered as a 1-D signal function.¹⁰⁻¹¹ As a consequence, each signal point corresponds to the spectral value of a particular wavelength in spectral dimension. With this interpretation, the MPCM can be implemented to capture progressive changes of spectral variation occurred in spectral wavelengths that are used to acquire the pixel.

One major advantage of using the PSSC is characterization of a spectral signature in progressive changes across its spectral channels. This unique feature cannot be accomplished by any “hard” coding-based spectral coding methods. Another advantage is the spectral profile of progressive changes produced for a signature can be used for various applications such as discrimination, classification, identification, etc. It is often the case that two signatures may be very similar in terms of spectral signature vector direction measured by spectral angle mapper (SAM),¹¹⁻¹² but in fact, they do have very different spectral profiles of progressive changes in a range of spectral channels. The PSSC provides such a profile for signature characterization. A third advantage is the change detection which is a major issue in land-cover remote sensing image classification and has been generally performed by temporal processing. The PSSC offers a different perspective in terms of change detection in spectral variation. A fourth advantage is that it can be viewed as a progressive implementation of a sequence of binary coding with a set of decreasing thresholds.

The proposed MPCM-PSSC provides a new look at how a SSC can be accomplished progressively for signature characterization. The idea is derived from the success of the MPCM in text detection for video images⁹ where the edges of text were detected more effectively in a progressive manner. Such progressive edge detection seems to be very useful in hyperspectral signature characterization. It generates a priority code that keeps track of progressive changes in spectral variation. The larger the change in a spectral wavelength is, the higher the priority of this particular wavelength is. Such an MPCM-PSSC generated priority code provides fingerprints of a spectral signature via priority code words assigned to each of spectral wavelengths. Here, the term of “code” is referred to as a code book which is made up of “code words” that are used for encoding. Another important advantage resulting from the MPCM-PSSC generated priority code is progressive decomposition of a spectral signature in according with the priority code words assigned to each of spectral wavelengths. The resulting progressive decomposition delineates a profile of progressive changes in spectral variation that can be used for discrimination and identification of a spectral signature, a feature that cannot be achieved by any spectral similarity measure. Furthermore, the MPCM-PSSC generated priority code can progressively reconstruct a spectral signature literally by the priority code words assigned to spectral wavelengths. This progressive signature reconstruction enables one to see how spectral changes are updated in order to recover the original signature from the MPCM-generated priority code. Most importantly, the MPCM-PSSC priority code can describe progressive transitions of spectral values from one spectral band to another via a simple coding scheme with a detailed profile of a spectral signature in terms of progressive changes in spectral variation across spectral wavelengths. Such capability makes the MPCM-PSSC unique. It distinguishes the MPCM-PSSC from a spectral similarity measure which can only measure the closeness or similarity between

two spectral signatures, not progressive spectral signature similarity across spectral wavelengths.

In order to facilitate analysis, a distinction between discrimination and identification suggested in Ref. [11] is also made clear in this paper. The former is performed among a set of signatures where one signature is discerned from another compared to the latter carried out by verifying a signature via a database (spectral library). Consequently, algorithms designed for discrimination and identification are slightly different. In particular, a threshold is generally required for signature discrimination to discriminate one signature from another. On the other hand, signature identification via a database can be performed directly by finding the one in the database that most matches the signature to be identified. In our proposed MPCM-PSSC, the signature discrimination and signature matching are measured by the priority code words using Hamming distance. Finally, computer simulations and real data experiments are conducted to demonstrate the utility of the MPCM-PSSC in applications of signature discrimination and identification.

The remainder of this paper is organized as follows. Section II describes the MPCM encoding scheme in detail. Section III presents an MPCM-based progressive spectral signature coding (MPCM-PSSC) where algorithms are developed for discrimination and identification. Section IV conducts extensive experiments to substantiate the proposed MPCM-PSSC. Section V summarizes the results and conclusions with some remarks.

2. MULTISTAGE PULSE CODE MODULATION (MPCM)

In this section, we present a new concept, called Multistage Pulse Coding Modulation (MPCM) that can be used for encoding spectral signatures in a progressive manner. The MPCM was originally developed for image progressive transmission and reconstruction.^{4,6} It can be

viewed as a progressive version of commonly used coding scheme in communications, Pulse Code Modulation (PCM).¹³ It expands the hard-decision PCM-based quantizer to a soft-decision quantizer in such a fashion that it allows PCM to have a non-decision region which passes on its decisions to next stage progressively. As a result, a decision can be refined stage-by-stage so as to improve quantization results. The idea of multistage coding is not new and can be found in Ref. [13] in which many references are available such as tree or residual coding^{2-3,13-14}, multi-resolution¹⁵, etc. Nevertheless, the MPCM is new and quite different from the reported references in the sense that it makes use of priority code words derived from multiple stages for progressive coding. The detailed idea of the MPCM can be described as follows.

A PCM-based coder is a quantizer, denoted by $Q(x)$ which is specified by a set of quantization levels $\{\Delta_k\}_{k=1}^M$ and a corresponding set of quantization thresholds $\{\tau_k\}_{k=1}^M$. It quantizes a signal function $x(n)$ according to

$$Q(x(n)) = \Delta_k \text{ if } x(n) \in [\tau_{k-1}, \tau_k) \quad (1)$$

where Δ_0 and Δ_M are initial conditions determined by the domain of the signal function $x(n)$. It is a hard decision-based quantizer, referred to as a hard quantization because $Q(x(n))$ must make a decision on the input $x(n)$ via Eq. (1) by assigning the quantization level Δ_k to $x(n)$. The proposed MPCM expands the $Q(x)$ in the sense that $x(n)$ in Eq. (1) is encoded by a sequence of M soft decision-based quantizers $\{Q_k(x(n))\}_{k=1}^M$ in multiple stages, referred to as soft quantizers, in a progressive manner compared to the hard decision made by one single value Δ_k in Eq. (1). Unlike the hard decision-based quantizer described in Eq. (1) which makes its binary decision on $x(n)$ by a single threshold interval $[\tau_{k-1}, \tau_k)$ for each quantization level Δ_k , $Q_k(x(n))$ makes its decision based on three threshold intervals, $(-\infty, -\Delta_k]$, $(-\Delta_k, \Delta_k)$ and $[\Delta_k, \infty)$ determined by

zero and its quantization level Δ_k where the interval $(-\Delta_k, \Delta_k)$ is designated as a no-decision threshold interval. More specifically, a soft quantizer $Q_k(x(n))$ derived from $Q(x(n))$ via the k -th quantization level Δ_k is defined by

$$Q_k(x(n)) = \begin{cases} -\Delta_k; & \text{if } x(n) \in (-\infty, -\Delta_k] \\ x(n); & \text{if } x(n) \in (-\Delta_k, \Delta_k) \\ \Delta_k; & \text{if } x(n) \in [\Delta_k, \infty) \end{cases} \quad (2)$$

where the soft quantizer $Q_k(x(n))$ passes its input $x(n)$ without making any decision when the input $x(n)$ falls in the region $x(n) \in (-\Delta_k, \Delta_k)$ as described in Fig. 1. The consequence of soft decisions comes from the inclusion of the no-decision interval, $(-\Delta_k, \Delta_k)$ in the quantizer $Q_k(x(n))$.

The MPCM takes advantage of such a soft quantizer $Q_k(x(n))$ specified by Eq. (2) to perform quantization progressively in multiple stages specified by $\{\Delta_k\}_{k=1}^M$, referred to as stage levels in the MPCM. Assume that $\{\Delta_k\}_{k=1}^M$ are strictly decreasing quantization levels, i.e., $\Delta_1 > \Delta_2 > \dots > \Delta_M > 0$. Therefore, the no decision-made outputs passed by the k -th soft quantizer by $Q_k(x(n))$ at stage k are further processed by the follow-up $(k+1)^{\text{st}}$ soft quantizer $Q_{k+1}(x(n))$ in next stage which uses a smaller quantization level, $\Delta_{k+1} < \Delta_k$ to refine its decision. In other words, instead of encoding $x(n)$ directly into Δ_k by Eq. (1), the $x(n)$ is actually encoded by M soft quantizers $\{Q_k(x(n))\}_{k=1}^M$ one at a time progressively using of M refined quantization levels. As a result of using a sequence of progressive soft quantizers $\{Q_k(x(n))\}_{k=1}^M$, $x(n)$ can be decomposed into a set of binary-valued stage components, denoted by $\{\hat{x}_k(n)\}_{k=1}^M$ where $\hat{x}_k(n) \in \{0,1\}$ for $1 \leq k \leq M$ so that the $x(n)$ can be approximated by the estimate of $x(n)$, $\hat{x}(n)$ by

$$\hat{x}(n) = \hat{x}_1\Delta_1 + \hat{x}_2\Delta_2 + \cdots + \hat{x}_M\Delta_M = \sum_{k=1}^M \hat{x}_k(n)\Delta_k \quad (3)$$

The key issue is how to find a desired set of M soft binary quantizers, $\{Q_k(x(n))\}_{k=1}^M$ for a given set of quantization levels $\{\Delta_k\}_{k=1}^M$ to produce an optimal M-block length binary code for Eq. (3) in approximation. In doing so, the soft quantizer using the quantization level Δ_k defined by Eq. (2) can be used for the k -th progressive soft quantizer in the MPCM defined by

$$Q_k(\varepsilon_{k-1}(n)) = \begin{cases} -\Delta_k; & \text{if } \varepsilon_{k-1}(n) \in (-\infty, -\Delta_k] \\ \varepsilon_{k-1}(n); & \text{if } \varepsilon_{k-1}(n) \in (-\Delta_k, \Delta_k) \\ \Delta_k; & \text{if } \varepsilon_{k-1}(n) \in [\Delta_k, \infty) \end{cases} \quad (4)$$

which takes the approximation error $\varepsilon_{k-1}(n) = x(n) - \sum_{j=1}^{k-1} \hat{x}_j(n)\Delta_j$ obtained at the $(k-1)$ st stage as its input. It should be noted that $\varepsilon_{k-1}(n) = x(n) - \sum_{j=1}^{k-1} \hat{x}_j(n)\Delta_j$ used in Eq. (4) is the approximation error obtained by a successive approximations using the binary code word $(\hat{x}_1(n)\hat{x}_2(n)\cdots\hat{x}_M(n))$ up to the $(k-1)$ st stage. The soft decision comes from the case that if $\varepsilon_{k-1}(n) \in (-\Delta_k, \Delta_k)$, $Q_k(\varepsilon_{k-1}(n)) = \varepsilon_{k-1}(n)$. A detailed implementation of the MPCM is described as follows. A generalized version of MPCM can be found in Refs. [4-6].

MPCM Encoding Algorithm for the n -th Signal Point, $x(n)$

1. Initial condition

Let $\{\Delta_k\}_{k=1}^M$ be a set of M stage levels which are used for MPCM and the initial approximation error $\varepsilon_0(n) = x(n) - \hat{x}(n-1)$ where $\hat{x}(n-1)$ is obtained by Eq. (3). Set $\hat{x}(0) = 0$ and $k = 1$.

2. At the k -th stage, three cases are considered for the k -th two-valued soft quantizer, Q_k defined by Eq. (4).

Case 1: If $\varepsilon_{k-1}(n) \geq \Delta_k$, then $Q_k(\varepsilon_{k-1}(n)) = \Delta_k$, $\hat{x}_k(n) = 1$ and set $\hat{x}_j(n) = 0$ for $k < j \leq M$. In this case, the priority codeword $c(n)$ assigned to $x(n)$ is $c(n) = k$. Its diagram is depicted in Fig. 2. Let $\varepsilon_M(n) = \varepsilon_{k-1}(n) - \sum_{j=k}^M \hat{x}_j(n)\Delta_j = \varepsilon_{k-1}(n) - \Delta_k$. Go to step 4.

Case 2: If $\varepsilon_{k-1}(n) \leq -\Delta_k$, then $Q_k(\varepsilon_{k-1}(n)) = -\Delta_k$, $\hat{x}_k(n) = 0$ and set $\hat{x}_j(n) = 1$ for $k < j \leq M$. In this case, the priority codeword $c(n)$ assigned to $x(n)$ is $c(n) = k$. Its diagram is depicted in Fig. 3. Let $\varepsilon_M(n) = \varepsilon_{k-1}(n) - \sum_{j=k}^M \hat{x}_j(n)\Delta_j = \varepsilon_{k-1}(n) - \sum_{j=k+1}^M \Delta_k$.

Go to step 4.

Case 3: If $-\Delta_k < \varepsilon_{k-1}(n) < \Delta_k$, then $Q_k(\varepsilon_{k-1}(n)) = \varepsilon_{k-1}(n)$ and $\hat{x}_k(n) = \hat{x}_k(n-1)$. Its diagram is depicted in Fig. 4. Go to step 3.

3. If $k < M$, let $k = k + 1$ and go to step 2. Otherwise, continue.
4. Go to the next sample, $(n+1)^{\text{st}}$ signal point, $x(n+1)$.

In the above MPCM encoding algorithm, the priority codeword is only assigned when a hard decision is made in a certain stage. When it occurs at stage k , the encoding for $x(n)$ is terminated and the priority code word for $x(n)$ is encoded as $c(n) = k$. In this case, the priority assigned to $x(n)$ is k , which indicates that there is a significant change in $x(n)$ at stage k . As a result, the higher the priority is, the bigger the change is, the smaller the index number of the stage is. That is, $c(n) = 1$ has the highest priority since there is a drastic change in stage 1 specified by the largest quantization level Δ_1 . To the contrary, $c(n) = M$ indicates that there only has a small change in stage M because the quantization level Δ_M is the smallest quantization level. Interestingly, an immediate advantage resulting from the MPCM encoding algorithm is that it allows one to decompose a signal sample $x(n)$ in multiple stages, i.e., M stages and its priority

code word indicates which stage the priority occurs where the signal sample makes a significant change.

Correspondingly, we also describe the MPCM decoding algorithm as follows. It decodes the n -th signal sample $x(n)$ based on the encoded priority code word $c(n)$ along with the previous decoded $\hat{x}(n-1)$ which is an approximation of $x(n-1)$ via Eq. (3). In contrast to the MPCM encoding algorithm which decomposes the n -th signal sample stage-by-stage in M stages, the MPCM decoding algorithm reconstructs the n -th signal samples stage-by-stage based on its priority code word $c(n)$.

MPCM Decoding Algorithm for $x(n)$

1. Initial condition:

Let $\{\Delta_k\}_{k=1}^M$ be a set of M stage levels which are used for MPCM. Set the initial condition as $\hat{x}(0) = 0$, and $\hat{x}(n-1)$ is the reconstruction of $x(n)$ which is unknown and can be expressed by $\{\hat{x}_k(n-1)\}_{k=1}^M$ as

$$\hat{x}(n-1) = \hat{x}_1(n-1)\Delta_1 + \hat{x}_2(n-1)\Delta_2 + \cdots + \hat{x}_M(n-1)\Delta_M \quad (5)$$

2. Input the encoded priority code word $c(n) = k$ for $x(n)$, in which case the priority of $x(n)$ occurs in stage k . Two cases are considered.

Case 1: if $\hat{x}_k(n-1) = 1$, then $\hat{x}_j(n) = \hat{x}_j(n-1)$ for $1 \leq j < k$, $\hat{x}_k(n) = 0$ and $\hat{x}_j(n) = 1$ for

$$k < j < M. \text{ In this case, } \hat{x}(n) = \sum_{j=1}^{k-1} \hat{x}_j(n-1)\Delta_j + \sum_{j=k+1}^M \Delta_j.$$

Case 2: if $\hat{x}_k(n-1) = 0$, then $\hat{x}_j(n) = \hat{x}_j(n-1)$ for $1 \leq j < k$, $\hat{x}_k(n) = 1$ and $\hat{x}_j(n) = 0$ for

$$k < j < M. \text{ In this case, } \hat{x}(n) = \sum_{j=1}^{k-1} \hat{x}_j(n-1)\Delta_j + \Delta_k.$$

In order to apply the MPCM to spectral signature coding we consider the spectrum of a

signature vector $\mathbf{r} = (r_1, r_2, \dots, r_L)^T$ as a 1-D signature vector with r_l being the spectral value of the l -th band. Let $\Delta(\mathbf{r}) = \max_l \{r_l\} - \min_l \{r_l\}$. The number of stages, M is then given by

$$M = \lceil \log_2 \Delta(\mathbf{r}) \rceil + 1 \quad (6)$$

with $\lceil x \rceil$ defined by the largest integer less than or equal to x . So, the stage levels $\{\Delta_k\}_{k=1}^M$ used in the MPCM is defined by

$$\Delta_k(\mathbf{r}) = 2^{-k} \Delta(\mathbf{r}) \text{ for } k = 1, 2, \dots, M. \quad (7)$$

In order to demonstrate the utility of the MPCM in spectral signature coding, two examples are provided for illustration.

The first example shows a progressive MPCM-encoded signal of a 1-dimensional chemical spectral data \mathbf{r} , methyl salicylate obtained from the web-book of National Institute of Standard Technology.¹⁶ It is shown in Fig. 5 and has 880 bands of spectral coverage 450-3966/cm.

In order for the MPCM to operate on this signal, the number of stages required for the MPCM encoding was calculated by Eq. (6) to be $M = 13$ stages. Since there are 13 stages, the stage levels obtained by Eq. (7) are $\Delta_k = \Delta(\mathbf{r}) / 2^k$ for $k = 1, 2, \dots, 13$. With the initial condition assumed to be $x(0) = 0$. Fig. 6 shows a graphical plot of the priority code words $c(n)$ for each of signal points $x(n)$ in Fig. 5 produced by the MPCM encoding algorithm with the x-axis and y-axis specified by signal points and their corresponding priority code words ranging from 1 to 13.

Using the MPCM encoded priority code words provided by Fig. 6, a 13-stage progressive signal components of the original signal in Fig. 5 can be decomposed stage-by-stage in Fig. 7. As we can see from Fig. 7, the MPCM encoding algorithm started with the largest stage level, $\Delta_1 = (2^{13} / 2) (\Delta(\mathbf{r}) / 2^{13}) = 4096 \times (\Delta(\mathbf{r}) / 2^{13})$ in stage 1, then began to reduce stage levels by half stage-by-stage to refine signal samples until it reached the last stage which is stage 13

specified by stage level $\Delta_{13} = \Delta(\mathbf{r}) / 2^{13}$.

In order to decode the signal of methyl salicylate, the MPCM-encoded priority code words in Fig. 6 was used as inputs and Fig. 8 shows the 13 decoded signal components of methyl salicylate progressively stage by stage for signature reconstruction along with the approximation error $\varepsilon_{13}(n)$.

Since it may not be trivial to fully understand how the MPCM works, the second example is provided by Table 1 for an illustrative purpose. It takes the first 20 signal points in Fig. 5 to walk through detailed stage-by-stage implementations of the MPCM encoding and decoding algorithms.

In Table 1, the 1st column lists the inputs specified by the first 20 signal points $\{x(n)\}_{n=1}^{20}$ with the initial condition specified by $x(0) = 0$. The 2nd column lists the values of predicted $\hat{x}(n)$ and predicted error $\varepsilon(n)$. The 3rd column lists all predicted values of signal components in 13 stages with stage levels specified by the largest stage level, $\Delta_1 = 2^{12} = 4096$ down to the smallest stage level, $\Delta_{13} = 2^0 = 1$. Finally, the last column produces the priority code words $\{c(n)\}_{n=1}^{20}$ for the first 20 signal points, $\{x(n)\}_{n=1}^{20}$.

Table 2 provides a stage-by-stage decoding process for signal reconstruction of the 20 MPCM signal samples encoded in Table 1 where the 1st column takes the priority code words from the output in Table 1 as the input to the MPCM decoder to decode the signal components in all 13 stages in the 2nd column. Finally, the last column of Table 2 outputs the predicted values of all the first 20 signal points of $\{x(n)\}_{n=1}^{20}$.

As concluding remarks in this section, two comment are noteworthy. In order to implement the MPCM-PSSC, a set of parameters, such as stage levels, $\{\Delta_k\}_{k=1}^M$ is required *a priori*. A

general approach to such selection is given by Eq. (7) which are empirically reasonable in our experiments. Another comment is that the computational complexity of the MPCM-PSSC is simple. As a matter of fact, it can be implemented as a real-time process for progressive transmission.

3. MPCM-BASED PROGRESSIVE SPECTRAL SIGNATURE CODING (MPCM-PSSC)

As recalled in the MPCM encoding algorithm, a signature vector $\mathbf{r} = (r_1, r_2, \dots, r_L)^T$ will be considered as a 1-D spectral signature where r_l is represented by one of the priority code words $\{c_k(\mathbf{r})\}_{k=1}^M$ taking values in $\{1, 2, \dots, M\}$. For example, $c_l(\mathbf{r})$ indicates the priority of r_l in the MPCM encoding and decoding. The smaller the number $c_l(\mathbf{r})$ is, the higher priority the $r_l(\mathbf{r})$ is for spectral encoding and decoding.

Next, we can further construct an M-dimensional priority unit vector associated with the priority code word $c_l(\mathbf{r})$ for MPCM-PSC as follows

$$\mathbf{c}_l(\mathbf{r}) = (c_{l1}(\mathbf{r}), c_{l2}(\mathbf{r}), \dots, c_{lM}(\mathbf{r}))^T \quad (8)$$

with $c_{lk}(\mathbf{r}) \in \{0,1\}$ and $\sum_{k=1}^M c_{lk}(\mathbf{r}) = 1$. The condition that $\sum_{k=1}^M c_{lk}(\mathbf{r}) = 1$ in Eq. (8) implies that $c_l(\mathbf{r})$ has only one “1” in its component and all zeros in its remaining components. It should be noted that the priority code word $c_l(\mathbf{r})$ takes the value in $\{1, 2, \dots, M\}$. Instead of using the priority code word $c_l(\mathbf{r})$ itself, we use its corresponding M-dimensional priority unit vector $\mathbf{c}_l(\mathbf{r})$ defined by Eq. (8) where the boldface of $c_l(\mathbf{r})$, $\mathbf{c}_l(\mathbf{r})$ indicates that it is the priority unit vector of the original scalar priority code word $c_l(\mathbf{r})$. As an example, for $M = 8$, $c_l(\mathbf{r})$ can take any of 8 values, 1, 2, 3, 4, 5, 6, 7, 8. In this case, the following 8-dimensional priority unit vectors derived from Eq. (8) can be used for spectral signature coding

$$\begin{aligned}
c_l(\mathbf{r}) = 1 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (1,0,0,0,0,0,0,0); & c_l(\mathbf{r}) = 2 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (0,1,0,0,0,0,0,0) \\
c_l(\mathbf{r}) = 3 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (0,0,1,0,0,0,0,0); & c_l(\mathbf{r}) = 4 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (0,0,0,1,0,0,0,0) \\
c_l(\mathbf{r}) = 5 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (0,0,0,0,1,0,0,0); & c_l(\mathbf{r}) = 6 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (0,0,0,0,0,1,0,0) \\
c_l(\mathbf{r}) = 7 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (0,0,0,0,0,0,1,0); & c_l(\mathbf{r}) = 8 &\Leftrightarrow \mathbf{c}_l(\mathbf{r}) = (0,0,0,0,0,0,0,1)
\end{aligned} \tag{9}$$

More specifically, if the priority code word $c_l(\mathbf{r})$ resulting from r_l is the priority, k , its M-dimensional priority unit vector $\mathbf{c}_l(\mathbf{r})$ is then specified by

$$\mathbf{c}_l(\mathbf{r}) = (\underbrace{0, 0, \dots, 0}_{1 \quad 2}, \underbrace{1, 0, \dots, 0}_{k-1 \quad k \quad k+1}, \underbrace{0, 0}_{M-1 \quad M})^T \tag{10}$$

where only one “1” occurs in the k -th component and represents its priority specified by the k -th stage. The advantage of using the M-dimensional priority unit vector, the position of “one” in Eq. (10) indicates the significance of its priority in the same manner that the bit position indicates the precision of the bit in a binary representation. Most importantly, we can use Eq. (10) and the Hamming distance to define a distance measure between two signature vectors $\mathbf{r} = (r_1, r_2, \dots, r_L)^T$ and $\mathbf{s} = (s_1, s_2, \dots, s_L)^T$ at the k -th stage via their corresponding M-dimensional priority unit vectors by

$$D_k(\mathbf{r}, \mathbf{s}) = \sum_{l=1}^L (c_{lk}(\mathbf{r}) \oplus c_{lk}(\mathbf{s})) \tag{11}$$

A. Spectral Discrimination

By virtue of Eq. (11), the similarity between two signature vectors \mathbf{r} and \mathbf{s} can be measured progressively. In other words, two signature vectors \mathbf{r} and \mathbf{s} are first measured by Eq. (11) in stage 1 via a prescribed stage threshold, say Δ_1 . If the distance $D_1(\mathbf{r}, \mathbf{s})$ is greater than Δ_1 , \mathbf{r} and \mathbf{s} will be declared to be distinct. Otherwise, the comparison between \mathbf{r} and \mathbf{s} is continued to proceed at stage 2. If the distance $D_2(\mathbf{r}, \mathbf{s})$ is greater than a prescribed stage threshold Δ_2 , \mathbf{r} and \mathbf{s} will be considered to be distinct signatures. Otherwise, a further comparison between \mathbf{r} and \mathbf{s} is continued on at stage 3, etc. The implementation of the MPCM-based progressive spectral

coding for target discrimination can be summarized as follows.

MPCM-PSSC Spectral Discrimination Algorithm

1. Let \mathbf{r} and \mathbf{s} be two spectral signature vectors to be discriminated.
2. Specify the number of stages, M , needed to be processed. If two signatures produce different stage numbers M_1 and M_2 , M is chosen as the minimum of M_1 and M_2 .
3. Determine the stage thresholds $\{\tau_k\}_{k=1}^M$ to be used for discrimination in each of M stages.
4. Apply the MPCM to \mathbf{r} and \mathbf{s} to generate their priority code words as described in Eq. (8) and expressed by Eq. (10).
5. Use Eq. (11) to measure the similarity between \mathbf{r} and \mathbf{s} progressively. For each stage k , we calculate the distance $D_k(\mathbf{r}, \mathbf{s})$ and compare it against the k -th stage threshold, τ_k . If $D_k(\mathbf{r}, \mathbf{s}) > \tau_k$, two pixel vectors \mathbf{r} and \mathbf{s} are declared to be distinct, and the process is terminated. Otherwise, repeat the same procedure until it reaches the last stage M . In this case, we check if $D_M(\mathbf{r}, \mathbf{s}) > \tau_M$.
6. If $D_M(\mathbf{r}, \mathbf{s}) > \tau_M$, two pixel vectors \mathbf{r} and \mathbf{s} are declared to be distinct, and the process is terminated.
7. If $D_M(\mathbf{r}, \mathbf{s}) \leq \tau_M$, the process is also terminated and output “no discrimination”, which declares \mathbf{r} and \mathbf{s} to be the same signature.

A key issue in implementing the MPCM-PSSC discrimination algorithm is the determination of an appropriate set of M stage thresholds for a signature. In doing so a simulated white Gaussian noise is added to the signature to achieve a certain level of signal-to-noise ratio (SNR). This SNR is determined by how much sensitivity we would like to have for a signature responding to its spectral variations.

B. Spectral Identification

The target identification studied in this section is different from spectral discrimination in Section III.A. While the target discrimination only discriminates one signature from another without performing any additional task such as detection, classification and identification, the target identification uses a given database (spectral library) Γ to identify an unknown target signature \mathbf{t} . Unlike spectral discrimination, the proposed spectral identification does not require stage thresholds.

MPCM-PSSC Spectral Identification Algorithm 1

1. Let Γ be a given database (spectral library) which is made up of p spectral signatures, $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_p$, i.e., $\Gamma = \{\mathbf{s}_h\}_{h=1}^p$ and \mathbf{t} be target spectral signature vector to be identified via the database Γ .
2. Specify the number of stages, M via Eq. (6), needed to be processed. For each signature \mathbf{s}_h , let M_h be the associated stage number. M is chosen as the minimum among M_1, M_2, \dots, M_p .
3. Determine stage thresholds for all M stages, $\{\tau_k\}_{k=1}^M$ for $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_p$.
4. Apply the MPCM to the target signature \mathbf{t} to generate its priority code. Set $k = 1$.
5. At the k -th stage, calculate the distance between \mathbf{t} and \mathbf{s}_h , $D_k(\mathbf{t}, \mathbf{s}_h)$ at stage k for $1 \leq h \leq p$ using Eq. (11). The \mathbf{t} is identified by \mathbf{s}_{h^*} with $h^* = \min_{1 \leq h \leq p} \{D_k(\mathbf{t}, \mathbf{s}_h)\}$. If there is a tie, the process is continued with those signatures that yield $\min_{1 \leq h \leq p} \{D_k(\mathbf{t}, \mathbf{s}_h)\}$ and continue.
6. If $k < M$, let $k \leftarrow k + 1$ and go to step 5. Otherwise, continue.

7. In this case, we reach the last stage M . The \mathbf{t} is identified by \mathbf{s}_{h^*} with $h^* = \min_{1 \leq h \leq p} \{D_M(\mathbf{t}, \mathbf{s}_h)\}$. If there is a tie at this final stage, the algorithm declares either “no match” or identifies \mathbf{t} as one of tied signatures.

The steps 5-7 in the above algorithm calculates the distance between \mathbf{t} and \mathbf{s}_h , $D_k(\mathbf{t}, \mathbf{s}_h)$ for each $1 \leq h \leq p$ stage-by-stage and makes a progressive decision to determine if there is a match between \mathbf{t} and \mathbf{s}_{h^*} for some h^* . There is no need of implementing stage thresholds as the way carried out by spectral discrimination. As an alternative, we can also replace the steps 5-7 to derive a second version of MPCM-PSSC target identification which postpones the decision until the last stage M by calculating the sum of stage distances between \mathbf{t} and \mathbf{s}_h in all M stages. In this case, the identification is to find the signature that yields the smallest sum.

MPCM-PSSC Spectral Identification Algorithm 2

The same first four steps used in MPCM-PSSC target identification algorithm 1.

5'. Compute $SUM_h = \sum_{k=1}^M D_k(\mathbf{t}, \mathbf{s}_h)$ and identify \mathbf{t} by \mathbf{s}_{k^*} with $h^* = \arg\{\min_{1 \leq h \leq p} \{SUM_h\}\}$, the signature that yields the smallest SUM_h . If there is a tie at this final stage, the algorithm declares either “no match” or identifies \mathbf{t} as one of tied signatures.

It should be noted that step 5' does not make its decision progressively. Instead, it makes its decision at the final stage, M , on the sum of all stage distances. Nevertheless, it does take advantage of progressive spectral signature changes occurred at each stage, each of which contributes its change to the sum.

4. CHEMICAL/BIOLOGICAL LABORATORY DATA EXPERIMENTS

The ability of the MPCM-PSSC in progressive signature decomposition and progressive

signature reconstruction was demonstrated in Figs. 7-8 and Tables 1 and 2. This and the following sections further demonstrate versatility of the MPCM-PSSC in other applications, spectral discrimination and identification. Two sets of data were used for experiments, laboratory data and real hyperspectral images. The laboratory data to be used in this section are chemical/biological spectral data available online at National Institute of Standards Technology (NIST)'s website.¹⁶ The data set has five 880-band chemical/biological spectral signatures shown in Fig. 9, which are methyl salicylate, pentanedione, propanoic acid, thiodiglycol, triethyl phosphate and heptanol. Since the selected data set for experiments was empirical and all the experiments conducted for this data set can be also applied to other data set in Ref. [16].

There are two reasons to select this data set. One is to demonstrate that the MPCM-PSSC has an application in chemical/biological defense. The other is to demonstrate that the MPCM-PSSC can be also used for ultraspectral signature characterization with a thousand of spectral channels. There are also some other applications such as hyperspectral laboratory data experiments that can be found in Ref. [17].

EXAMPLE 1 (Spectral discrimination)

In order to perform spectral discrimination using the MPCM-PSSC, we need to determine appropriate thresholds for each stage that are implemented by MPCM-PSSC stage by stage. For each signature we create a noise-corrupted signature with signal-to-noise ratio (SNR) 30:1 where the SNR was defined in ref. [18] as the ratio of 50% reflectance to noise standard deviation. Using the methyl salicylate in Fig. 5 as an example, the spectral signature represented by the methyl salicylate is denoted by $\mathbf{r} = (r_1, r_2, \dots, r_{880})^T$. Then a noise corrupted methyl salicylate signature denoted by $\tilde{\mathbf{r}} = (\tilde{r}_1, \tilde{r}_2, \dots, \tilde{r}_{880})^T$ can be obtained by adding a white Gaussian noise to each band to achieve the signal-to-noise (SNR) = 30:1. Finally, the MPCM is applied to both the pure

methyl salicylate signature with no noise and the 30:1 SNR noise corrupted methyl salicylate signature to obtain their respective MPCM priority code word for band l , $\mathbf{c}_l = (c_{l1}, c_{l2}, \dots, c_{lM})^T$ and $\tilde{\mathbf{c}}_l = (\tilde{c}_{l1}, \tilde{c}_{l2}, \dots, \tilde{c}_{lM})^T$ with $M = 13$. Then the k -th stage threshold Δ_k is obtained by

$$\tau_k = \sum_{l=1}^{880} c_{lk} \oplus \tilde{c}_{lk} \quad (13)$$

Table 3 tabulates all the stage thresholds $\{\tau_k\}_{k=1}^M$ for each of five signatures, methyl salicylate, pentanedione, propanoic acid, thiodiglycol, triethyl phosphate and heptanol which are denoted by $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4$ and \mathbf{s}_5 .

It should be noted that the total number of stages, $M = 13$ is determined by Eq. (6). As long as $\{\tau_k\}_{k=1}^M$ is determined, the discrimination process starts with the stage threshold in stage 1. If the distance between two signatures in stage 1 is greater than the threshold, the two signatures are declared to be distinct and discrimination process is terminated. Otherwise, it implies that two signatures cannot be discriminated in stage 1 and the discrimination process is then passed on to stage 2 where the distance between two signatures in stage 2 is calculated and compared to the threshold at stage 2. If the distance at stage 2 is greater than the threshold, the process is terminated. Otherwise, the same procedure is repeated again until the last stage is reached.

Since the stage thresholds produced by one signature generally are different from those produced by another signature, the discrimination threshold is then determined by the minimum of the two different stage thresholds, that is, $\min\{\tau_i(\text{signature 1}), \tau_i(\text{signature 2})\}$. Table 4 shows the results where the stage thresholds in Table 3 were used for discrimination and the numbers highlighted (shaded) underneath each stage were the stages that two signatures were discriminated. As we can see from Table 4, all the five signatures can be discriminated in stage 1.

EXAMPLE 2 (Spectral identification)

In this example, we further demonstrate the utility of the MPCM-PSSC in spectral identification via a database (spectral library) Γ which consists of the five signatures in Fig. 9. For each target signature \mathbf{t} , 60% abundance fraction was simulated while the other four signatures sharing the remaining 40% abundance fraction with each of 10% abundance fraction. Five different admixtures were generated by a fixed mixing composition (0.6,0.1,0.1,0.1,0.1) of the five signatures. When one of $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4$ and \mathbf{s}_5 was designated as a target signature, say \mathbf{s}_1 , a mixed signature \mathbf{s} was then generated by mixing 0.6 of \mathbf{s}_1 with abundance fraction of 0.1 from each of the other four signatures $\mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4$ and \mathbf{s}_5 . Table 5 tabulates a progressive spectral identification process for such a mixed signature \mathbf{s} where the signature \mathbf{s} was quickly identified by the target signature $\mathbf{t} = \mathbf{s}_1$ immediately by Algorithm 1 in the first stage as well as by Algorithm 2 correctly.

Similar experiments were also performed by changing the designated target signature \mathbf{t} from \mathbf{s}_1 to $\mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4$ and \mathbf{s}_5 for two spectral identification algorithms. Tables 6-9 tabulate their respective spectral progressive identification results. All the four mixed signatures are correctly identified by both Algorithm 1 and Algorithm 2. This experiment indicated that \mathbf{s}_4 as \mathbf{s}_5 are very similar to each other in terms of spectral variation. Algorithm 1 had difficulty with identification until stage 2.

As a concluding remark, the abundance fraction of the target signature \mathbf{t} has impact on the performance of the MPCM-PSSC in identification. If the abundance fraction was greater than 60%, the MPCM-PSSC improved significantly its performance. Otherwise, its performance deteriorates as the abundance fraction gradually diminishes. In the following real image experiments, we will further demonstrate that the MPCM-PSSC can still perform effectively

when the estimated abundance fraction of a subpixel target is above 40%.

5. REAL IMAGE HYPERSPECTRAL EXPERIMENTS

The second data set used for experiments was a real HYperspectral Digital Imagery Collection Experiment (HYDICE) image shown in Fig. 10(a), which has size of 64×64 pixel vectors with 15 panels in the scene. Within the scene there also has a large grass field background, a forest on the left edge and a barely visible road running on the right edge of the scene. Low signal/high noise bands: bands 1-3 and bands 202-210; and water vapor absorption bands: bands 101-112 and bands 137-153 were removed. The spatial resolution is 1.56m and spectral resolution is 10nm. There are 15 panels located in the center of the grass field and are arranged in a 5×3 matrix as shown in Fig. 10(b) which provides the ground truth map of Fig. 10(a). Each element in this matrix is a square panel and denoted by \mathbf{p}_{ij} with row indexed by $i = 1, 2, 3, 4, 5$ and column indexed by $j = 1, 2, 3$. For each row 5×3 , the three panels $\mathbf{p}_{i1}, \mathbf{p}_{i2}, \mathbf{p}_{i3}$ were painted with the same material but have three different sizes. For each column $j = 1, 2, 3$, the five panels $\mathbf{p}_{1j}, \mathbf{p}_{2j}, \mathbf{p}_{3j}, \mathbf{p}_{4j}, \mathbf{p}_{5j}$ have the same size but were painted with five different materials. It should be noted that the panels in rows 2 and 3 are made by the same material, but with different paints, so did the panels in rows 4 and 5. Nevertheless, they were still considered as different materials. The sizes of the panels in the first, second and third columns are $3\text{m} \times 3\text{m}$, $2\text{m} \times 2\text{m}$ and $1\text{m} \times 1\text{m}$ respectively. So, the 15 panels have five different materials and three different sizes. Fig. 10(b) shows the precise spatial locations of these 15 panels where red pixels (R pixels) are the panel center pixels and the pixels in yellow (Y pixels) are panel pixels mixed with background. The 1.56m-spatial resolution of the image scene suggests that most of the 15 panels are one pixel in size except that $\mathbf{p}_{21}, \mathbf{p}_{31}, \mathbf{p}_{41}, \mathbf{p}_{51}$ which are two-pixel panels, denoted by

p_{211} , p_{221} , p_{311} , p_{312} , p_{411} , p_{412} , p_{511} , p_{521} . Since the size of the panels in the third column is $1m \times 1m$, they cannot be seen visually from Fig. 10(a) due to the fact that its size is less than the 1.56m pixel resolution.

Fig. 10(c) plots the five panel spectral signatures $\{\mathbf{p}_i\}_{i=1}^5$ obtained from Fig. 10(b), where the i -th panel signature, denoted by P_i was generated by averaging R pixels in row i . These panel signatures will be used to represent target knowledge of the panels in each row.

Two scenarios will be conducted for experiments based this 15-panel HYDICE scene. One is spectral discrimination among the five panel signatures, \mathbf{p}_1 , \mathbf{p}_2 , \mathbf{p}_3 , \mathbf{p}_4 and \mathbf{p}_5 . The other is to identify the 15 panels unsupervisedly using only knowledge obtained directly from the data.

EXAMPLE 3 (Spectral discrimination)

Like Example 1, the spectral discrimination was performed by the MPCM-PSSC where the number of stages required for MPCM-PSSC was calculated by Eq. (6) to be $M = 13$ and the stage levels $\{\Delta_k\}_{k=1}^{13}$ were obtained by Eq. (7). In order to implement MPCM-PSSC algorithms, we also need to determine an appropriate set of stage thresholds.

Using the same way conducted in Example 1, the desired set of stage thresholds $\{\tau_k\}_{k=1}^{13}$ were obtained in Table 10 by Eq. (13) using noise-corrupted signatures with signal-to-noise ratio (SNR) set to 30:1 as variation of signature tolerance.

Table 11 tabulates the discrimination results obtained by the MPCM-PSSC among the five panel signatures $\{\mathbf{p}_i\}_{i=1}^5$ in Fig. 10(c). As shown in Table 11, \mathbf{p}_1 and \mathbf{p}_2 are more similar each other than other three panel signatures since the discrimination could be accomplished in stage 2 in terms of spectral variation compared to other signature discrimination which was discriminated in stage 1.

EXAMPLE 4 (Spectral identification)

The experiments conducted in this example are very interesting and offer several intriguing results and observations. It was designed to identify the 19 R panel pixels, p_{ij} in Fig. 10(b) by the MPCM-PSSC. Since the panel pixels p_{13} , p_{23} , p_{33} , p_{43} , p_{53} have size of $1m \times 1m$ which is smaller than pixel size, their abundance fractions present in single pixels can be at most $1/(1.56)^2 = 0.4109$ which can be interpreted as approximately 50% of pixel size. As a result, the performance in identification of these subpixel panels can be expected to be very challenging and difficult. On the other hand, due to its very high spatial and spectral resolution the spectral variations of image pixels in this HYDICE scene can be very subtle and sensitive. Therefore, using the five panel signatures $\{\mathbf{p}_i\}_{i=1}^5$ in Fig. 10(c) as a data base may not be appropriate. Instead, a more effective data base must be obtained in an unsupervised means directly from data. In doing so, the result of the 34 target pixels generated directly from the scene by an unsupervised fully constrained least squares (UFCLS) method developed in Refs. [11,19] was used to form a desired data base Γ . Among these 34 generated target pixels there were five panel pixels identified to correspond to the five distinct panel signatures, $\{\mathbf{p}_i\}_{i=1}^5$. Table 12 tabulates the results produced by the MPCM-PSSC using Algorithm 1 and Algorithm 2 for target identification along with the abundance fractions of the 19 R pixels estimated by the fully constrained least squares method in Ref. [19] where an identification error is highlighted by shade.

According to Table 12, Algorithm 1 yielded the best performance in the sense that it only missed identification when the panels, p_{13} , p_{212} , p_{33} , p_{412} , p_{43} , p_{53} with estimated abundance fractions less than 0.3821. Algorithm 2 also made 6 identification errors, but it seemed that these misidentifications had no clear tie to the abundance fractions as the way Algorithm 1 did. For example, it correctly identified p_{212} whose abundance is only 0.3141, but it misidentified the p_{32} whose abundance is 0.5343. Compared to Algorithm 2, the SAM and the spectral information

divergence (SID)¹¹⁻¹² not only made the same 6 identification errors as did Algorithm 2, but also made two more additional errors, which are panel pixels p_{511} , p_{52} with abundance fractions, 0.7203 and 0.7789.

This experiment showed that the MPCM-PSSC performed more effectively than a pixel-based spectral similarity measure such as SAM and SID in Table 12. It should be noted that real target pixels in Table 12 were compared against the five panel signatures $\{\mathbf{p}_i\}_{i=1}^5$ for analysis.

It is interesting to note that if the five panel signatures $\{\mathbf{p}_i\}_{i=1}^5$ in Fig. 10(c) were directly used for identification, the results were reported in Ref. [17] and were not as good as the results in Table 12 that were produced by using the real target pixels in Table 12. This is primarily due to the fact that the signatures $\{\mathbf{p}_i\}_{i=1}^5$ obtained by averaging R panel pixels are not real pixels. As a result, the signature variations of real target pixels have been compromised. The MPCM-PSSC seemed to remedy such deficiency by capturing subtle spectral variations in multiple stages which can dictate changes in subtle difference encountered in real data as shown in Table 12.

As a final comment, it should be noted that the 34 target pixels used in this experiment were shown in Ref. [19] to sufficiently enough to include target pixels that represent the five distinct panel spectral signatures. However, it did not imply that it required at least 34 target pixels to do so. There may have some unsupervised target detection and classification algorithms that can generate a fewer number of target pixels than 34 but still include pixels that can represent all the desired five panel signatures. In this case, these generated target pixels can be used as a data base as well. As expected, the conclusion drawn from Table 12 will remain unchanged.

6. CONCLUSIONS

This paper introduces a new concept of progressive spectral signature coding (PSSC) for

hyperspectral signature characterization. It is derived from a technique, called multistage pulse code modulation (MPCM) that was previously developed for progressive image reconstruction and edge detection. Unlike the commonly used spectral signature coding which performs coding with hard decision, the proposed PSSC characterizes a hyperspectral signature in a sequence of soft decisions in multiple stages to produce a profile of progressive changes in spectral variation of the spectral signature. The idea of the MPCM-based PSSC (MPCM-PSSC) is to use a sequence of soft decision-based quantizers to generate a priority code for a hyperspectral signature which can be used to prioritize the signature values across its spectral coverage whose priorities are specified by stage levels implemented in various stages. Such a priority code allows one to decompose and reconstruct a hyperspectral signature progressively in accordance with the priorities assigned to spectral signature values in wavelengths. As a result, a profile of progressive changes in spectral variation can be generated for a hyperspectral signature and can be further used to dictate subtle differences in spectral characterization. In order to substantiate the utility of the proposed MPCM-PSSC applications in spectral discrimination and identification are considered and investigated. Experiments are also conducted to demonstrate unique features of the MPCM-PSSC in hyperspectral signature characterization such as progressive spectral changes, progressive signature decomposition and progressive signature reconstruction which cannot be found in any spectral signature coding.

ACKNOWLEDGMENTS

The first author would like to thank support received from his NRC (National Research Council) senior research associateship sponsored by the US Army Soldier and Biological Command, Edgewood Chemical and Biological Center (ECBC).

References

1. K.H. Tzou, "Progressive image transmission: a review and comparison of techniques," *Optical Engineering*, **26**(7), 581-589 (1987).
2. L. Wang and M. Goldberg, "Progressive image transmission using vector quantization on images in pyramid form," *IEEE Transaction on Communications*, **37**(12), 1339-1349 (1989)
3. L. Wang and M. Goldberg, "Progressive image transmission by transform coefficient residual error quantization," *IEEE Transaction on Communications*, **36**(1), 75-87 (1988).
4. Yaqi Cheng, *Multistage Pulse Code Modulation (MPCM)*, MS thesis, Department of Electrical Engineering, University of Maryland, Baltimore County, Baltimore, MD (1993).
5. C.-I Chang, Y. Cheng, M.L.G. Althouse, L. Zhang and J. Wang, "Multistage image coding: a top-down gray-level triangle method," *Proc. International Symposium on Spectral Sensing Research (ISSSR)*, Kauai, Hawaii, pp. 497-511, Sep. 15-20 (1992).
6. C.-I Chang and Y. Cheng and M.L.G. Althouse, "Chemical vapor detection using multistage predictive coding," *Proc. Scientific Conference on Chemical Defense Research*, CRDEC, Aberdeen Proving Ground, MD, pp. 909-915, Nov. 17-20 (1992).
7. C.-J. Chang, C.-I Chang and M.-L. Chang, "Subband multistage predictive coding," *Proc. Int. Conf. on Signal Processing '93/Beijing*, Beijing, China, pp. 783-787, Oct. 26-30, (1993).
8. C.-I Chang, Y. Cheng, J. Wang, M.L.G. Althouse and M.L. Chang, "Progressive edge extraction using multistage predictive coding," *Proc. 1994 International Symposium on Speech, Image and Neural Networks*, Hong Kong, pp. 57-60, April 14-16 (1994).
9. Y. Du, C.-I Chang and P. Thouin, "An Automatic System for text detection in single video images," *J. Electronic Imaging*, **12**(3), 410-422 (2003).

10. R.A. Schwengerdt, *Remote Sensing, Models and Methods for Image Processing*, 2nd ed., New York, Academic Press (1997).
11. C.-I Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, Kluwer Academic/Plenum Publishers (2003).
12. C.-I Chang, "An information theoretic-based approach to spectral variability, similarity and discriminability for hyperspectral image analysis," *IEEE Trans. on Information Theory*, **46**(5), 1927-1932 (2000).
13. A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, 1992.
14. B.H. Juang and A.H. Gray, Jr., "Multiple stage vector quantization for speech coding," *Int. Conf. on ASSP*, vol. 1, 597-600, Paris (1982).
15. S. Mallat, "A theory for multiresolution signal decomposition," *IEEE Pattern Analysis and Machine Intelligence*, **11**(7), 674-693, (1989).
16. webbook.nist.gov/chemistry.
17. C.-I Chang, J. Wang, F. D'Amico and J.O. Jensen, "Multistage pulse code modulation for progressive spectral signature coding," *Chemical and Biological Standoff Detection*, SPIE, pp. 252-261, 27-31 October (2003).
18. J. Harsanyi and C.-I Chang, "Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach," *IEEE Trans. on Geoscience and Remote Sensing*, **32**(4), 779-785 (1994).
19. Heinz, D. and C.-I Chang (2001), "Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery," *IEEE Trans. on Geoscience and Remote Sensing*, **39**(3), pp. 529-545 (2001).

Tables

Table 1. The first 20 MPCM-encoded signal samples in Fig. 5 using 13 stages for signal decomposition

MPCM ENCODING ALGORITHM																	
INPUT		PREDICT ION		SIGNAL COMPONENTS											OUTPUT		
n	$x(n)$	$\hat{x}(n)$	$\varepsilon(n)$	\hat{x}_1	\hat{x}_2	\hat{x}_3	\hat{x}_4	\hat{x}_5	\hat{x}_6	\hat{x}_7	\hat{x}_8	\hat{x}_9	\hat{x}_{10}	\hat{x}_{11}	\hat{x}_{12}	\hat{x}_{13}	$c(n)$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	367	256	111	0	0	0	0	1	0	0	0	0	0	0	0	0	5
2	144	255	-111	0	0	0	0	0	1	1	1	1	1	1	1	1	5
3	33	127	-94	0	0	0	0	0	0	1	1	1	1	1	1	1	6
4	108	111	-3	0	0	0	0	0	0	1	1	0	1	1	1	1	9
5	70	95	-25	0	0	0	0	0	0	1	0	1	1	1	1	1	8
6	106	96	10	0	0	0	0	0	0	1	1	0	0	0	0	0	8
7	59	63	-4	0	0	0	0	0	0	0	1	1	1	1	1	1	7
8	119	64	55	0	0	0	0	0	0	1	0	0	0	0	0	0	7
9	157	128	29	0	0	0	0	0	1	0	0	0	0	0	0	0	6
10	162	160	2	0	0	0	0	0	1	0	1	0	0	0	0	0	8
11	198	192	6	0	0	0	0	0	1	1	0	0	0	0	0	0	7
12	233	224	9	0	0	0	0	0	1	1	1	0	0	0	0	0	8
13	240	240	0	0	0	0	0	0	1	1	1	1	0	0	0	0	9
14	223	223	0	0	0	0	0	0	1	1	0	1	1	1	1	1	8
15	245	224	21	0	0	0	0	0	1	1	1	0	0	0	0	0	8
16	290	256	34	0	0	0	0	1	0	0	0	0	0	0	0	0	5
17	388	384	4	0	0	0	0	1	1	0	0	0	0	0	0	0	6
18	516	512	4	0	0	0	1	0	0	0	0	0	0	0	0	0	4
19	591	576	15	0	0	0	1	0	0	1	0	0	0	0	0	0	7
20	665	640	25	0	0	0	1	0	1	0	0	0	0	0	0	0	6

Table 2. The first 20 MPCM decoded signal points for signal reconstruction in Fig. 5 with 13 stages

MPCM DECODING ALGORITHM															
INPUT		SIGNAL COMPONENTS													OUTPUT
n	$c(n)$	\hat{x}_1	\hat{x}_2	\hat{x}_3	\hat{x}_4	\hat{x}_5	\hat{x}_6	\hat{x}_7	\hat{x}_8	\hat{x}_9	\hat{x}_{10}	\hat{x}_{11}	\hat{x}_{12}	\hat{x}_{13}	$\hat{x}(n)$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	5	0	0	0	0	1	0	0	0	0	0	0	0	0	256
2	5	0	0	0	0	0	1	1	1	1	1	1	1	1	255
3	6	0	0	0	0	0	0	1	1	1	1	1	1	1	127
4	9	0	0	0	0	0	0	1	1	0	1	1	1	1	111
5	8	0	0	0	0	0	0	1	0	1	1	1	1	1	95
6	8	0	0	0	0	0	0	1	1	0	0	0	0	0	96
7	7	0	0	0	0	0	0	0	1	1	1	1	1	1	63
8	7	0	0	0	0	0	0	1	0	0	0	0	0	0	64
9	6	0	0	0	0	0	1	0	0	0	0	0	0	0	128
10	8	0	0	0	0	0	1	0	1	0	0	0	0	0	160
11	7	0	0	0	0	0	1	1	0	0	0	0	0	0	192

12	8	0	0	0	0	0	1	1	1	0	0	0	0	0	224
13	9	0	0	0	0	0	1	1	1	1	0	0	0	0	240
14	8	0	0	0	0	0	1	1	0	1	1	1	1	1	223
15	8	0	0	0	0	0	1	1	1	0	0	0	0	0	224
16	5	0	0	0	0	1	0	0	0	0	0	0	0	0	256
17	6	0	0	0	0	1	1	0	0	0	0	0	0	0	384
18	4	0	0	0	1	0	0	0	0	0	0	0	0	0	512
19	7	0	0	0	1	0	0	1	0	0	0	0	0	0	576
20	6	0	0	0	1	0	1	0	0	0	0	0	0	0	640

Table 3. 13 stage thresholds for 5 signatures in Fig. 9 with SNR 30:1

stage	1	2	3	4	5	6	7	8	9	10	11	12	13
s_1	2	3	7	16	23	28	40	47	58	58	54	48	47
s_2	1	9	6	19	29	40	61	61	83	96	94	83	81
s_3	2	3	6	10	21	27	38	49	50	57	70	66	64
s_4	1	2	9	26	27	52	92	110	150	150	140	80	130
s_5	0	0	1	5	8	18	20	30	37	55	55	58	87

Table 4. Discrimination among 5 signatures in Fig. 9 using the stage thresholds in Table 3

stage	1	2	3	4	5	6	7	8	9	10	11	12	13
$s_1 - s_2$	10	32	49	80	94	120	170	190	190	200	160	110	120
$s_1 - s_3$	12	32	53	72	100	120	150	180	180	180	190	120	140
$s_1 - s_4$	12	30	57	98	100	150	200	160	180	170	150	87	150
$s_1 - s_5$	10	28	48	69	88	120	140	160	180	180	170	130	260
$s_2 - s_3$	6	22	36	54	88	110	140	160	180	200	180	140	170
$s_2 - s_4$	6	20	42	78	82	140	180	160	190	190	160	110	170
$s_2 - s_5$	4	18	27	45	66	100	140	130	180	210	160	130	280
$s_3 - s_4$	8	16	44	78	88	130	160	170	160	180	170	130	170
$s_3 - s_5$	6	14	35	39	80	98	96	140	160	170	180	170	270
$s_4 - s_5$	6	14	39	67	76	120	150	130	170	180	160	110	280

Table 5. Spectral identification for a mixed signature, $s = 0.6t + 0.1s_2 + 0.1s_3 + 0.1s_4 + 0.1s_5$ with $t = s_1$

stage	1	2	3	4	5	6	7	8	9	10	11	12	13	SUM
s_1	6	24	46	80	110	120	140	170	140	130	130	70	68	1234
s_2	12	34	47	88	100	130	170	170	190	190	150	99	120	1500
s_3	14	34	55	92	110	130	150	170	170	170	180	130	150	1555
s_4	14	30	61	110	110	170	200	160	170	160	140	79	160	1564
s_5	12	30	46	77	95	130	140	150	180	170	160	120	270	1580

Table 6. Spectral identification for a mixed signature, $s = 0.1s_1 + 0.6t + 0.1s_3 + 0.1s_4 + 0.1s_5$ with $t = s_2$

stage	1	2	3	4	5	6	7	8	9	10	11	12	13	SUM
s_1	10	32	53	87	120	150	180	190	170	170	160	96	98	1516
s_2	0	16	22	59	98	130	140	160	150	190	150	110	100	1325
s_3	6	20	42	65	100	130	150	190	160	180	170	130	160	1503
s_4	6	18	48	83	110	160	190	180	170	160	150	95	150	1520
s_5	4	16	31	54	96	120	130	150	170	180	170	120	290	1531

Table 7. Spectral identification for a mixed signature, $\mathbf{s} = 0.1\mathbf{s}_1 + 0.1\mathbf{s}_2 + 0.6\mathbf{t} + 0.1\mathbf{s}_4 + 0.1\mathbf{s}_5$ with $\mathbf{t} = \mathbf{s}_3$

stage	1	2	3	4	5	6	7	8	9	10	11	12	13	SUM
s_1	14	34	57	93	100	160	160	180	170	170	140	110	100	1488
s_2	8	30	40	79	92	120	170	160	200	190	160	130	130	1509
s_3	6	18	24	59	84	130	130	150	160	160	150	140	120	1331
s_4	10	24	46	95	92	150	180	160	170	170	140	110	150	1497
s_5	8	24	39	58	78	120	130	140	170	180	140	140	260	1487

Table 8. Spectral identification for a mixed signature, $\mathbf{s} = 0.1\mathbf{s}_1 + 0.1\mathbf{s}_2 + 0.1\mathbf{s}_3 + 0.6\mathbf{t} + 0.1\mathbf{s}_5$ with $\mathbf{t} = \mathbf{s}_4$

stage	1	2	3	4	5	6	7	8	9	10	11	12	13	SUM
s_1	12	34	55	110	130	140	190	180	180	150	150	95	130	1556
s_2	6	24	36	83	110	130	190	160	190	190	160	110	130	1519
s_3	8	22	40	85	110	130	160	160	170	160	170	120	170	1505
s_4	6	14	42	87	110	150	170	130	150	140	130	72	96	1297
s_5	6	20	31	74	110	110	140	130	150	170	160	130	270	1501

Table 9. Spectral identification for a mixed signature, $\mathbf{s} = 0.1\mathbf{s}_1 + 0.1\mathbf{s}_2 + 0.1\mathbf{s}_3 + 0.1\mathbf{s}_4 + 0.6\mathbf{t}$ with $\mathbf{t} = \mathbf{s}_5$

stage	1	2	3	4	5	6	7	8	9	10	11	12	13	SUM
s_1	10	26	54	72	120	150	180	170	190	160	160	120	140	1552
s_2	4	20	35	56	92	130	180	160	180	210	170	130	170	1537
s_3	6	16	41	58	94	120	150	160	180	170	170	150	160	1475
s_4	6	16	47	72	94	150	190	140	170	160	150	98	170	1463
s_5	2	6	28	37	88	110	130	130	150	150	150	130	230	1341

Table 10. Stage thresholds for five panel signatures with SNR 30:1

stage	1	2	3	4	5	6	7	8	9	10	11	12	13
\mathbf{p}_1	2	1	5	10	14	24	23	16	16	16	9	4	4
\mathbf{p}_2	2	2	14	9	13	23	24	15	13	11	10	4	4
\mathbf{p}_3	1	4	6	8	16	19	21	13	11	11	7	6	5
\mathbf{p}_4	3	6	3	11	15	21	23	20	16	12	7	7	3
\mathbf{p}_5	3	6	4	10	16	19	21	18	13	10	7	4	3

Table 11. Discrimination among five panel signatures using the stage thresholds in Table 10

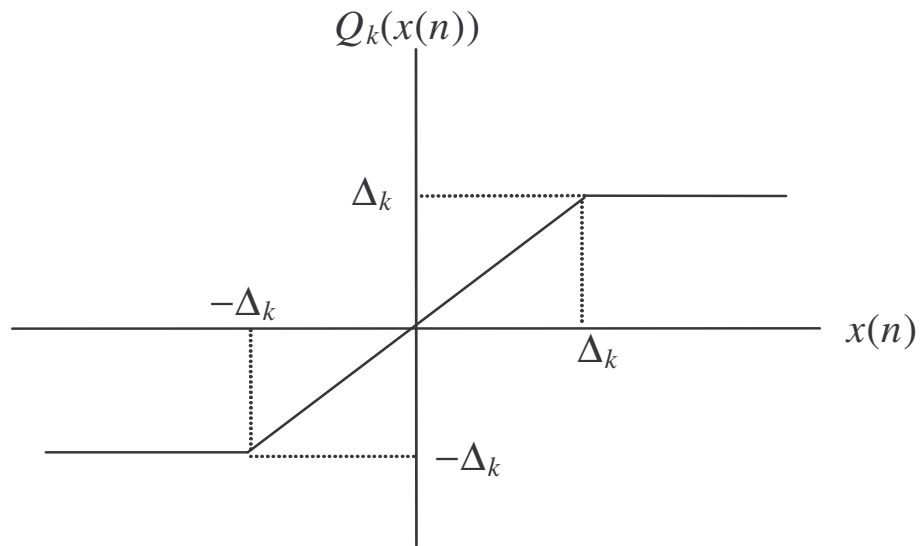
stage	1	2	3	4	5	6	7	8	9	10	11	12	13
p₁- p₂	2	2	10	13	20	32	30	24	29	21	16	4	5
p₁- p₃	4	4	17	23	28	29	31	26	24	22	11	6	5
p₁- p₄	4	8	8	22	26	37	32	26	22	23	10	6	4
p₁- p₅	4	8	11	22	32	34	34	21	13	16	12	2	5
p₂- p₃	2	2	13	12	24	35	31	22	21	9	11	6	4
p₂- p₄	4	8	14	23	28	33	36	22	21	10	10	8	3
p₂- p₅	4	8	17	23	28	30	30	25	28	13	8	4	4
p₃- p₄	6	10	19	27	30	34	31	12	20	13	9	10	5
p₃- p₅	6	10	22	29	34	35	29	29	17	16	7	6	6
p₄- p₅	4	8	3	6	16	25	24	29	21	13	4	6	1

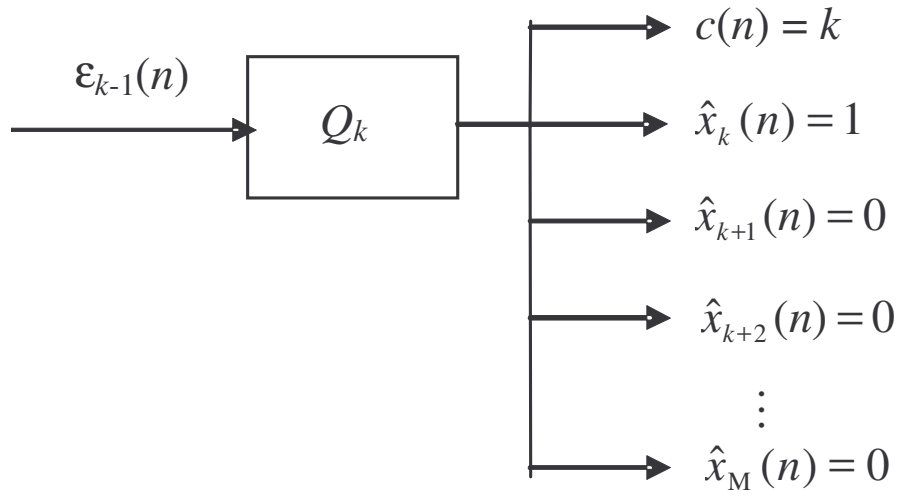
Table 12. Identification of 19 R panel pixels in Fig. 10(a)

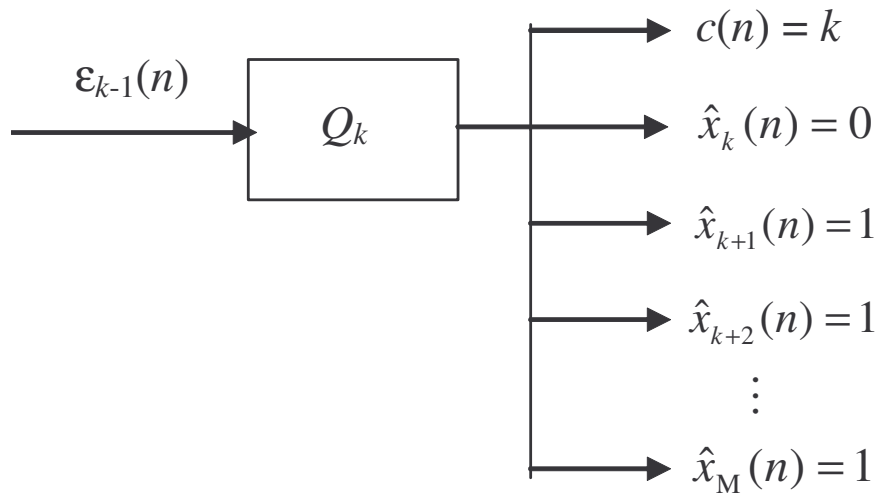
panel pixels	Algorithm 1	Algorithm 2	SAM/ SID	abundance fractions estimated by FCLS
p ₁₁	p₁	p₁	p₁	1
p ₁₂	p₁	p₂	p₂	0.4098
p ₁₃	p₃	p₂	p₂	0.0499
p ₂₁₁	p₂	p₂	p₂	0.5255
p ₂₂₁	p₃	p₂	p₂	0.3141
p ₂₂	p₂	p₂	p₂	0.6917
p ₂₃	p₂	p₂	p₂	0.4221
p ₃₁₁	p₃	p₃	p₃	0.8647
p ₃₁₂	p₃	p₃	p₃	1
p ₃₂	p₃	p₂	p₂	0.5343
p ₃₃	p₂	p₂	p₂	0.3285
p ₄₁₁	p₄	p₄	p₄	1
p ₄₁₂	p₅	p₄	p₄	0.3821
p ₄₂	p₄	p₄	p₄	0.7034
p ₄₃	p₂	p₂	p₂	0.2242
p ₅₁₁	p₅	p₅	p₄	0.7203
p ₅₂₁	p₅	p₅	p₅	1
p ₅₂	p₅	p₅	p₄	0.7789
p ₅₃	p₂	p₂	p₂	0.1466

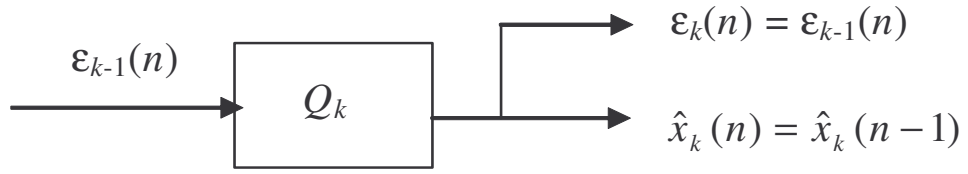
A List of Figures

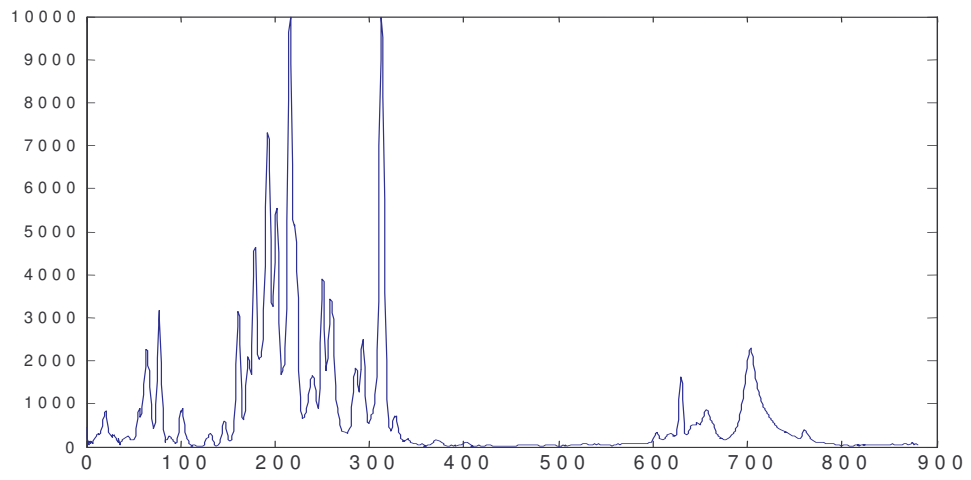
1. Figure 1. A soft quantizer $Q_k(x(n))$ described by Eq. (2)
2. Figure 2. Case 1 for MPCM encoding algorithm
3. Figure 3. Case 2 for MPCM encoding algorithm
4. Figure 4. Case 3 for MPCM encoding algorithm
5. Figure 5. Spectral signature of methyl salicylate, \mathbf{r}
6. Figure 6. Graphical plot of priority code words for the signal of methyl salicylate in Fig. 5
7. Figure 7. MPCM-encoded progressive spectral signatures of methyl salicylate
8. Figure 8. A progressive stage-by stage decoded spectral signatures of methyl salicylate from the priority code words in Fig. 6 along with the approximation error $\varepsilon_{13}(n)$.
9. Figure 9. Five spectral signatures of chemical data from NIST
10. Figure 10. (a) A HYDICE panel scene which contains 15 panels; (b) Ground truth map of spatial locations of the 15 panels; (c) Spectral signatures of \mathbf{p}_1 , \mathbf{p}_2 , \mathbf{p}_3 , \mathbf{p}_4 and \mathbf{p}_5

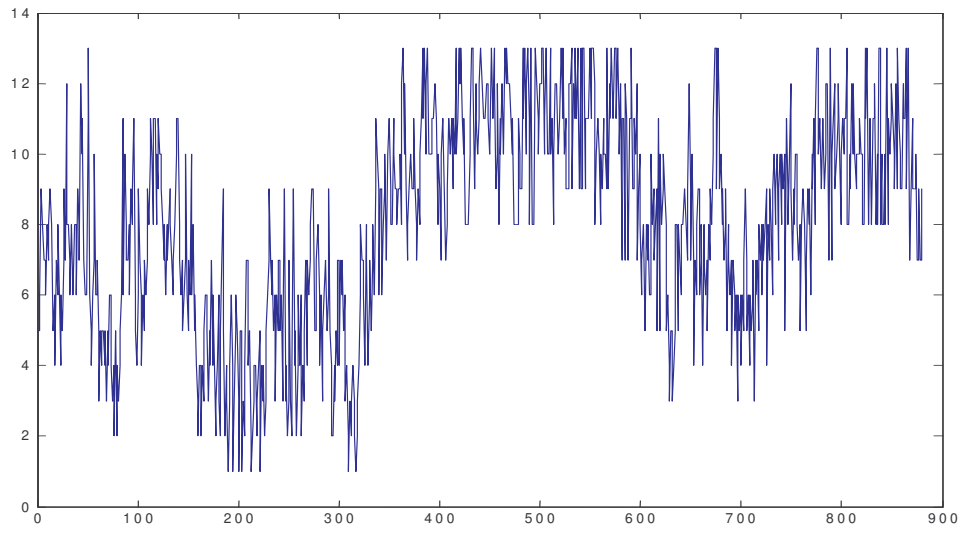


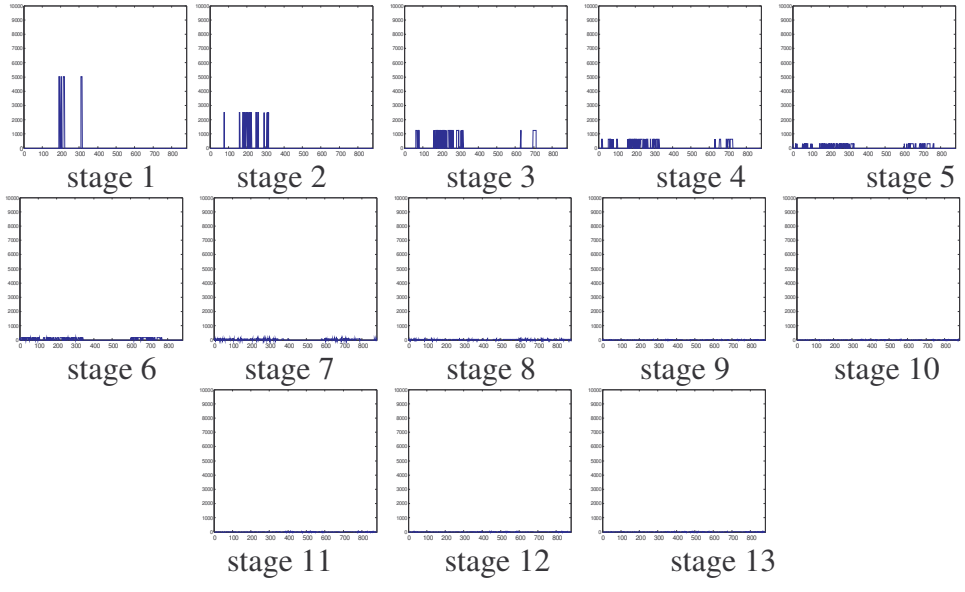


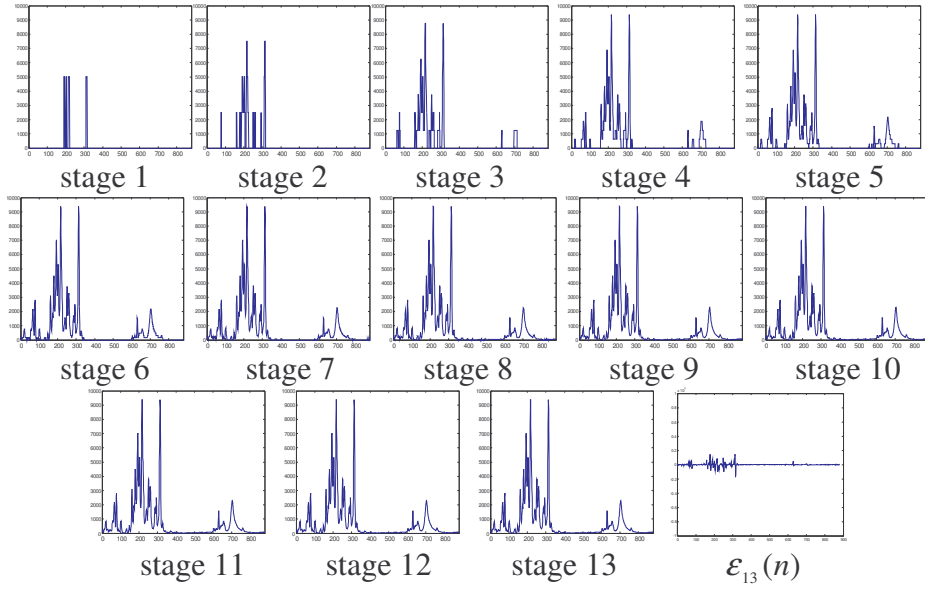


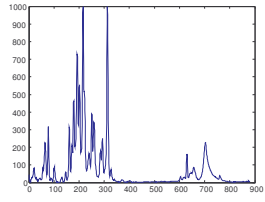




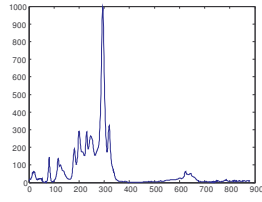




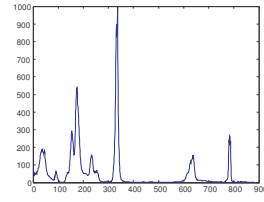




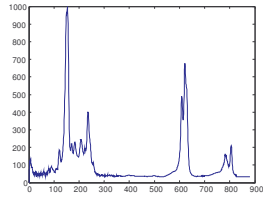
s₁: Methyl Salicylate



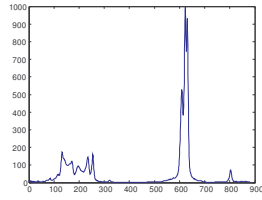
s₂: Pentanedione



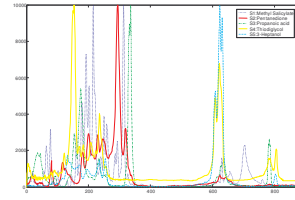
s₃: Propanoic acid



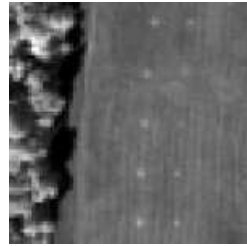
s₄: Thiodiglycol



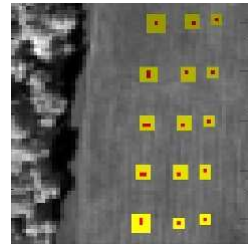
s₅: 3-Heptanol



5 signatures

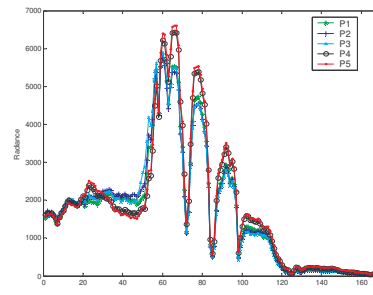


(a)



(b)

- P11, P12, P13
- P211, P22, P23
P221
- P311, P312, P32, P33
- P411, P412, P42, P43
- P511, P52, P53
P521



(c)